

# *Двумерный массив*

Учитель информатики  
МБОУ «Марковская СОШ»  
Репникова С.А.

В практической деятельности человека часто используются таблицы и списки.

**Элементы списков и номера строк и столбцов всегда нумеруются**

- Список учеников в журнале;
- Список среднесуточной температуры месяца;
- Таблица умножения.

1. Иванов
2. Петров
3. Сидоров
4. Кошкин
5. Мишкин

1.  $10^0$
2.  $12^0$
3.  $12^0$
4.  $13^0$
5.  $14^0$
6.  $14^0$
7.  $12^0$
8.  $15^0$
9.  $16^0$
10.  $14^0$

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60

В языках программирования подобные типы данных описываются с помощью массивов.

---

# Массив

это набор однотипных данных, к которым можно обратиться с помощью единственного имени

Каждое значение в массиве называется **элементом**.

Номер элемента в списке называется **индексом**.

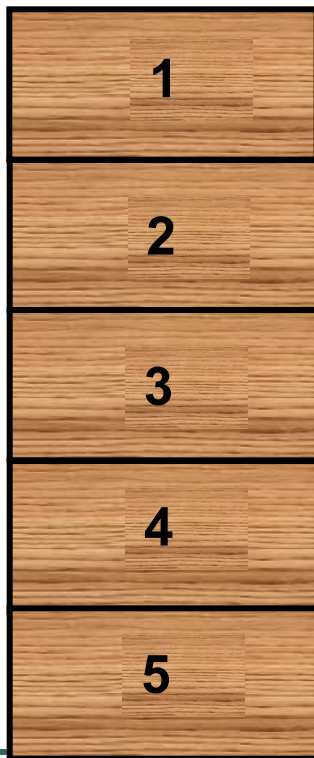
Элементы представляют собой символьные или числовые значения, которые можно использовать в выражениях ЯП ПАСКАЛЬ

# Одномерный массив

В качестве иллюстрации можете представить себе шкаф, имеющий множество нумерованных ящиков.

---

*Шкаф*



Шкаф – это **массив**.

Ящики – это ***индексы***.

Содержимое ящиков –  
***элементы массива***.

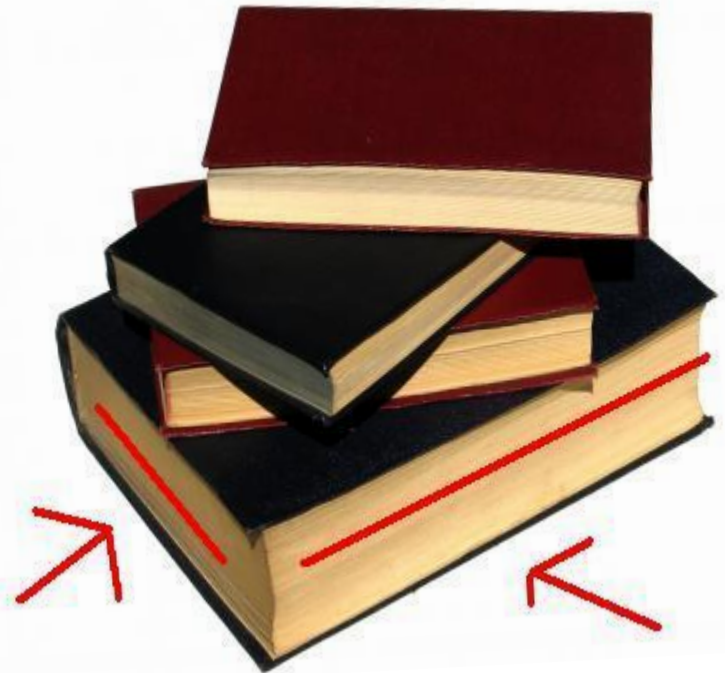
Доступ к содержимому конкретного ящика (элемента массива) осуществляется после указания шкафа – имени массива – и номера ящика – индекса массива.

Обычную книгу  
можно считать своего  
рода массивом.

---

Почему?

Ответ: книга состоит из множества однотипных элементов – страниц, у каждой страницы есть номер (индекс), все страницы объединены под одним названием (название книги)



Футбольную команду  
можно считать  
«массивом». Почему?

---

*Ответ: команда состоит из  
нескольких людей, у каждого из  
них есть номер (индекс).*



Дом также можно  
считать массивом.  
Почему?

Ответ: дому  
соответствует один  
почтовый адрес  
(город, улица, номер).  
Элементами дома  
можно считать  
квартиры, у каждой из  
которых есть номер  
(индекс).



Шахматную доску  
можно считать  
массивом. Почему?

Доска состоит из  
клеток, каждая  
клетка  
обозначается  
буквой и цифрой  
(двойной индекс).





# Двумерный массив

---

Зрительный зал кинотеатра – двумерный массив.

Название кинотеатра – **ИМЯ массива.**

Ряд, место - **индексы.**

Человек на

определенном месте

– **элемент массива.**



---

Массив представляет собой совокупность данных одного типа с общим для всех элементов именем.

Имя массива состоит из букв(буквы) латинского алфавита.

## Характеристики массива:

---

- Тип – общий тип всех элементов массива;
- Размерность (ранг) – количество индексов массива;
- Диапазон изменения индекса (индексов), определяет количество элементов в массиве

## Двумерный массив

---

Массив, который состоит из нескольких строк и нескольких столбцов называется двумерным массивом.

Его элементы нумеруются двумя индексами – номером строки и номером столбца соответственно.

(Аналогом явл. ваше место в кинотеатре, где номер строки массива – это ряд, а номер столбца массива – это ваше место в ряду)

---

Двумерные массивы располагаются в памяти ПК по строкам: сначала все элементы первой строки, затем элементы второй строки и т.д.

Двумерный массив называют также матрицей.

---

Если количество строк и количество столбцов в массиве одинаково, то такой массив называется квадратной матрицей.

Наиболее часто индексы массива – это переменные типа **integer**.

---

При обращении к элементу массива указывается имя массива, а в квадратных скобках через запятую индексы (номер строки, номер столбца) этого элемента массива.

Например  $A[3,2]$  мы обратились к элементу массива  $A$ , который располагается в третьей строке во втором столбце.

---

Самый простой способ описания массива в программе – это объявить переменную в разделе описания переменных **var** с использованием зарезервированного слова **array**( т.е. массив):

**var** имя массива: **array**[нижняя граница индекс **1**..  
Верхняя граница индекс**1**, нижняя граница индекс**2**..  
верхняя граница индекс **2**] **of** тип элементов;



## Объявление массива

---

Var

a:array [1..4, 1..3] of integer;

Const

n=4; m=3;

Var

a:array [1..n, 1..m] of integer;

## Например:

---

Пусть в памяти ПК расположена таблица чисел:

1	2	3	4
5	6	7	8
9	10	11	12

Тогда описание этого массива:

```
Var b:array[1..3,1..4] of integer;
```

{12 элементов массива – целые числа}

## Внимание:

---

Реальное количество элементов в массиве может быть меньше, чем указано при описании, но ни в коем случае не должно быть больше.

## Заполнение массива данными

---

Для заполнения двумерного массива данными используются вложенные циклы с параметом: For...

Внешний цикл организуется по номеру строки, а внутренний цикл – по номеру столбца. Значения элементов массива можно задавать вводом данных с клавиатуры.

## Ввод значений

---

```
For i:= 1 to n do
  for j:=1 to m do begin
    write ('A[',i,j,']=');
    readln(a[i,j]); {с клавиатуры}
  end;
```

## Ввод значений

---

```
For i:= 1 to n do  
  for j:=1 to m do  
    a[i,j]:= i*j; {по формуле}
```

## Ввод значений

---

For i:= 1 to n do

  for j:=1 to m do

    a[i,j]:= random(100); *{заполнение массива случайными числами}*

## Вывод элементов двумерного массива по строкам и столбцам:

---

```
For i:=1 to 3 do
```

```
begin
```

```
For j:=1 to 2 do
```

```
Write(B[i,j]:5); {для того, чтобы числа выводились равными столбцами}
```

```
Writeln;
```

```
End;
```



## Действия над элементами двумерного массива:

---

Пусть имеется массив  $A$ , в котором  $n$  строк и  $m$  столбцов.

- 1) **Вычисление суммы элементов каждой строки.**

Результатом является массив с именем  $D$ , состоящий из  $n$  сумм элементов строк.

For  $i:=1$  to  $n$  do

Begin

$S:=0$ ;

For  $j:=1$  to  $m$  do  $s:=s+a[i,j]$ ;

$D[i]:=s$ ;

End;

## Приемы программирования

{нахождение суммы всех элементов по столбцам}

For j:= 1 to m do

begin

S:=0;

for i:=1 to n do

S:=S+A[i,j];{сумма значений в столбце}

writeln(S);

end;

## Приемы программирования

---

{нахождение суммы всех элементов}

S:=0;

For i:= 1 to n do

    for j:=1 to m do

        S:=S+A[i,j];

## Приемы программирования

---

{нахождение произведения всех элементов}

p:=1;

For i:= 1 to n do

  for j:=1 to m do

    p:=p\*A[i,j];

## Приемы программирования

---

{нахождение среднего арифметического всех элементов}

S:=0;

For i:= 1 to n do

  for j:=1 to m do

    S:=S+A[i,j];{сумма значений в столбце}

  end;

Sr:=S/(n\*m);

writeln('среднее арифметическое-'Sr:7:2);

## Поиск минимального элемента всей матрицы.

---

- Переменная  $min$  используется для хранения значений минимального элемента,  $K$  – номер строки,  $L$  – номер столбца, где он находится

**Min:=A[1,1];** { поиск начинаем с 1 элемента}

**K:=1; L:=1;**

**For i:=1 to n do**

**For j:=1 to m do**

**If A[i,j]< min then**

**Begin**

**min:=A[i,j];**

**k:=i; L:=j;**

**End;**



## Приемы программирования

---

{поиск наибольшего элемента в массиве}

Max:=a[1,1];

For i:= 1 to n do

    for j:=1 to m do

        if a[i,j]>max then max:=a[i,j];

    writeln('наибольшее значение-',max);

## Приемы программирования

---

{найти наименьшее значение из максимумов в каждой строке}

Min:=10000;

For i:=1 to n do begin

    Max:=a[i,1];

    for j:=1 to m do begin

        if a[i,j]>max then max:=a[i,j];

    end;

    if max<min then min:=max;

End;

Writeln ('min=',min);

## Решите задачу:

Сформировать  
таблицу  
Пифагора  
(таблица  
умножения в  
виде  
прямоугольной  
таблицы вида:

1	2	3	4	5	...	9
2	4	6	8	10		18
3	6	9	12	15		27
4	8	12	16	20		36
5	10	15	20	25		45
...						
9	18	27	36	45		81